Node-level Performance of Adaptive Resolution in `ls1 mardyn`

Jose Alfonso Pinzon Escobar[1][0000−0003−0420−4230], Alex Hocks[2][0009−0002−0666−5233], and Philipp Neumann[3][0000−0001−8604−8846]

[1] Chair for High Performance Computing,
Helmut Schmidt University, Germany
`jose.pinzon@hsu-hh.de`
[2] Technical University of Munich
`alex.hocks@tum.de`
[3] High Performance Computing & Data Science, University of Hamburg, Germany
IT-Department, DESY, Germany
`philipp.neumann@desy.de`

Abstract. In this work we present a node-level performance analysis of an adaptive resolution scheme (AdResS) implemented in `ls1 mardyn` . This is relevant in simulations involving a very large number of particles or long timescales, because it lowers the computational effort required to calculate short-range interactions in molecular dynamics. An introduction to AdResS is given, together with an explanation of the coarsening technique used to obtain an effective potential for the coarse molecular model, i.e., the Iterative Boltzmann Inversion (IBI). This is accompanied by details of the implementation in our software package, as well as an algorithmic description of the IBI method and the simulation workflow used to generate results. This will be of interest for practitioners. Results are provided for a pure Lennard-Jones tetrahedral molecule coarsened to a single site, validated by verifying the correct reproduction of structural correlation functions, e.g. the radial distribution function. The performance analysis builds upon a literature-driven methodology, which provides a theoretical estimate for the speedup based on a reference simulation and the size of the full particle region. Additionally, a strong scaling study was performed at node level. In this sense, several configurations with vertical interfaces between the resolution regions are tested, where different resolution widths are benchmarked. A comparison between several linked cell traversal routines, which are provided in `ls1 mardyn` , was performed to showcase the effect of algorithmic aspects on the adaptive resolution simulation and on the estimated performance.

Keywords: Molecular dynamics · Adaptive resolution · Iterative Boltzmann Inversion

1 Introduction

Molecular dynamics has become a prevalent tool for analyzing diverse phenomena at the atomistic detail. One of its limitations is that for a high number

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

of molecules, the runtime can increase tremendously. Most of the computation time is spent calculating inter- and intramolecular forces between the particles. One way of decreasing this computational effort, without much loss of physical meaning in the simulations, is to use Adaptive Resolution Simulations or Schemes (AdResS) [11, 5, 12].Where a multi-site molecule is replaced by a single-site equivalent particle located at the center of mass.

In this work, we present results for adaptive resolution using `ls1 mardyn` [8]. Our focus is on what performance trade offs occur between different widths of the resolution regions and the provided traversals. Further details on `ls1 mardyn` and the used cell traversal schemes used in this work can be found in [2, 7]. Our benchmarks are limited to the speedup as presented in [3], due to simplicity and higher focus on computational performance. Alternatively, a more physics-oriented approach can be found in [1].

In section 2, the fundamental theory of AdResS is presented, together with the details on how to generate an effective potential for coarsened molecules. It is followed by section 3, where implementation details and algorithmic aspects are described. Finally, section 4 gives validation results based on a tetrahedral molecule scenario together with a performance analysis based on the previously mentioned factors. The conclusion presents our final observations and a perspective on further work to be done in `ls1 mardyn` .

2 Theoretical Background

2.1 Adaptive Resolution

Adaptive resolution was first presented for a Lennard-Jones fluid in [11], where a fine-grain, molecular representation composed of several sites is coarsened to a single interaction site placed at the center of mass (COM). Therefore, the fine-grain model is called full particle (FP) and the coarsened is referred to as the coarse-grain (CG) model. As show in fig. 1, these molecular representations do not interact with each other directly, but indirectly through a mixed model involving both the multi- and single site representations. The widths of the resolution regions are determined heuristically, making sure they are at least larger than the cutoff radius to avoid inter-resolution interactions.

In the hybrid (HY) region, the forces are given by

$$\boldsymbol{F}_{\alpha\beta} = w(x_\alpha)w(x_\beta)\boldsymbol{F}^{\mathrm{FP}}_{\alpha\beta} + [1 - w(x_\alpha)w(x_\beta)]\boldsymbol{F}^{\mathrm{CG}}_{\alpha\beta}, \tag{1}$$

where $w(x_\alpha)$ is a weight function whose value depends on the molecules within the HY region, and $\boldsymbol{F}^{\mathrm{FP}}_{\alpha\beta}$ and $\boldsymbol{F}^{\mathrm{CG}}_{\alpha\beta}$ are the forces corresponding to the interactions between the FP and CG sites, respectively.

The weight function is defined as in [17],

$$w(x) = \begin{cases} 1 & \text{in FP} \\ \cos^2\left[\frac{\pi}{2\,d_{HY}}\,(x_{\mathrm{HY}}\,)\right] & \text{in HY} \\ 0 & \text{in CG} \end{cases} \tag{2}$$

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

with the widths of the HY region given by d_{HY}. The coordinate x_{HY} refers to the position of the COM of a given molecule over the x-axis, measured with respect to the center of the domain. It can be seen that the function $w(x)$ has zero value in the CG region and one in the FP, while in the HY region it is smooth and monotonic.

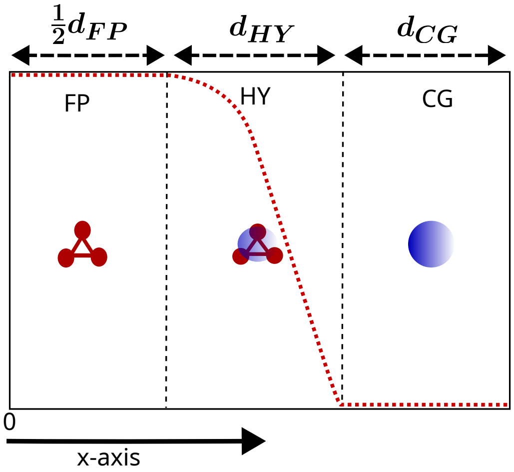


Fig. 1. Schematic representation of an AdResS simulation. The different resolution regions are labeled as hybrid (HY), coarse-grain (CG), and full-particle (FP). Additionally, the respective molecular level of resolution is show for each region. Their respective widths are given. The dotted (red) line describes the used weight function. The x-coordinate shows that the domain is centered at the origin.

2.2 Effective Potential

After generating an equilibrated phase space of only FP molecules, it is necessary to obtain a potential that governs the interactions between CG particles in the domain. With this purpose, the Iterative Boltzmann Inversion (IBI) proposed in [13] is used, which is used to obtain an effective potential, U^{CG}. The application of the force field derived from this potential result in the preservation of a predetermined structural correlation function from the original FP reference domain. A commonly used approach, is to obtain U^{CG} that reproduces the radial distribution function (RDF), which provides a statistical description of how the density varies with respect to the distance to a reference particle, of the equilibrated FP phase space up to a predefined tolerance. In [11] it is mentioned that for phases of low density, an accurate first guess is defined as

$$U_0^{\mathrm{CG}}(r) = -T\kappa_B \ln\left(g_{\mathrm{ref}}(r)\right), \tag{3}$$

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

which is called the potential of mean force (PMF), where T is the ensemble temperature and κ_B is Boltzmann's constant, see [10, 6]. It provides a good approximation to the ideal gas behavior at low densities and a good starting point for the iterative process. Then, the iterative process uses a correction step given by

$$U_{i+1}^{\mathrm{CG}}(r) = U_i^{\mathrm{CG}}(r) + \alpha T \, \kappa_B \ln \left(\frac{g_i(r)}{g_{\mathrm{ref}}(r)} \right), \tag{4}$$

where the second term can be understood as a correction $\Delta U_i(r)$ to the previously found potential. The correction $\Delta U_i(r)$ can be multiplied by a step-size $\alpha \in [0, 1]$, as in [6].

Moreover, the reference RDF, $g_{\mathrm{ref}}(r)$, is found as the average of an FP simulation at the same thermodynamic state through standard techniques such as binning. In this sense, every update of the potential represents a simulation where the current RDF, $g_i(r)$ is computed. Therefore, after several IBI-updates, the potential converges to the required effective potential $U^{\mathrm{CG}}(r)$. Further information on coarse-grain potentials can be found in [14].

3 Algorithmic Aspects

Within this work, the AdResS features were implemented in the open-source software `ls1 mardyn` [8], using its plugin functionality. The simulations are separated into three different phases. First, an equilibration phase, where an FP-only domain comprised of N molecules is equilibrated for a given number of steps. The reference RDF, $g_{\mathrm{ref}}(r)$, is subsequently measured and imported to the next phase. The equilibration is followed by an effective potential phase, which is based on the experiments presented in [6], the effective potential is computed using a CG-only simulation and IBI. The initial molecular positions are based on the molecular COMs of the equilibrated phase space generated in the equilibration phase. The output of this simulation is $U^{\mathrm{CG}}(r)$, which describes the interactions between CG pairs. Finally, the AdResS simulation phase is performed, as sketched in fig. 1, where the HY region is an interface between the CG and FP molecules. The converged effective potential is used for the interactions involving CG molecules.

3.1 IBI Implementation

The mathematical aspects of the computation of the effective potential via the IBI method were described in section 2.2. Here, an algorithmic description is attempted while illustrating practical aspects for practitioners, thus algorithm 1 is included to provide more insight regarding IBI implementation in `ls1 mardyn` . Further details can be also found in [14, 10].

In algorithm 1, the steps used for an implementation of the IBI method are given. Starting from a CG-only domain, the reference RDF is used to compute

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

the initial guess, U_0^{CG}, for the potential and forces. This RDF has been previously computed from an equilibrated FP phase space. In order to compute an effective potential that reproduces the $g_{\mathrm{ref}}(r)$, an only-CG particles domain is used. Therefore, the CG particles are assigned initial positions corresponding to the COM of the multi-site molecules of the equilibrated phase space.
The simulation will run for N total steps or until convergence of the potential is reached. Updates of the potential require an averaged $g_i(r)$, therefore a step stride N_{IBI} controls how often a correction is computed and applied. As mentioned in section 2.2, a step size α can be used to decrease numerical overshoots of the correction. Additionally, a tolerance parameter ϵ_{IBI} is given. The convergence is measured using the fit parameter given in [6] as

$$e_{\mathrm{RDF}} = 1 - \frac{\sum_{b=0}^{bins} \left(|g_i(r_b) - g_{\mathrm{ref}}(r_b)|\right)}{\sum_{b=0}^{bins} \left(|g_i(r_b)| + |g_{\mathrm{ref}}(r_b)|\right)}, \tag{5}$$

where r_b are the centers of the profiling bins, *bins* is the total number of used bins in the profiler, and a e_{RDF} closer to one means better convergence.

Algorithm 1 Iterative Boltzmann Inversion

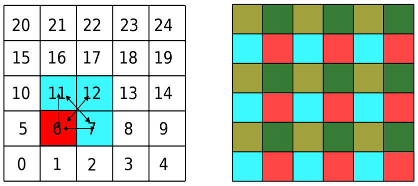
```
Require: $N$,$N_{\mathrm{IBI}}$,$\alpha$,$\epsilon_{IBI}$, equilibrated phase space
Read in $g_{\mathrm{ref}}(r)$
Coarse grain phase space
Compute initial guess $U_0^{\mathrm{CG}}$ with eq. (3)
for $i = 0....N$ do
    Average $g_i(r)$
    Compute intermolecular forces using $\nabla_x U_{CG}(r(x))$
    if $N \mod N_{\mathrm{IBI}} = 0$ then
        if $(1.0 - e_{\mathrm{RDF}}) \leq \epsilon_{\mathrm{IBI}}$ then
            Return effective potential $U^{\mathrm{CG}}(r)$
            End simulation
        end if
        Potential update using eq. (4)
    end if
end for
```

The data points of the discrete vectors, $g_i(r)$ and $g_{\mathrm{ref}}(r)$, are located at the centers of the bins used to measured the correlation functions. A linear interpolation is used to find values within the discrete correlation functions, and central finite differences are used to compute the gradient of the CG potential.

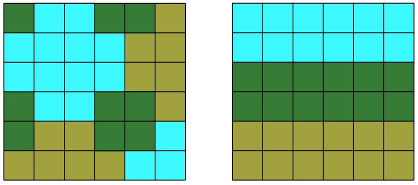
3.2 AdResS Performance in `ls1 mardyn`

The computation of the short-range forces using `ls1 mardyn` is implemented with the linked cells data structure, which divides the domain into equally sized cubic cells. The edge size of these cells is usually set equal to the cutoff radius,

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4



(a) `c08` base step (b) `c08` coloring pattern



(c) `c04` coloring (d) `sliced` coloring

Fig. 2. OpenMP-based parallelization schemes in ls1 mardyn. (a) Compact stencil of pairwise force evaluations between cell pairs. (b) Coloring scheme leveraging compact stencil from (a). Cells of same color can be interacted with the stencil pattern simultaneously, whereas a synchronization/barrier is required after finishing each color. Depicted for four colors in 2D (eight colors in 3D, respectively). (c) Alternative coloring approach using only 3 colors in 2D (four colors in 3D, respectively). (d) Slicing approach, in which slices of cells are traversed with the compact stencil simultaneously. Locking mechanisms at the interfaces between the slices circumvent racing conditions. For details, see [15].

r_c. Additional halo cells are introduced, either for the implementation of the minimum image convention when simulating periodic boundary conditions or for communication between processes after domain decomposition. The pairwise forces then imply the traversal of all the cells and their neighbors, to find all the molecular pairs within r_c.

Moreover, an efficient computation of the aforementioned forces requires that this traversal is performed in an equally optimal manner. In `ls1 mardyn` this is achieved by the implementation of node-level partitioning schemes, based on OpenMP parallelization, as introduced in [16] for the `c08` and `sliced` traversals and in [15] for the `c04`. These routines are reproduced for ease of convenience in fig. 2. By optimizing the total number of neighbor cells traversed for each base cell, the `ls1 mardyn` traversals avoid race conditions and allow to actively apply Newton's third law equality $F_{ij} = -F_{ji}$ for a force from particle j acting on particle i to halven the overall number of force calculations.

As sketched in fig. 1, an adaptive resolution simulation involves three resolution levels of the same molecule. In algorithmic terms, this implies that there might be cells which include a mixture of representations. This fact can have a positive or negative impact on the performance of the simulation depending on which traversal is used, thus making this aspect non-trivial. As a result, the selection of the traversing routine and the size of the resolution regions have a fundamental role in the performance of an AdResS-enabled simulation.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

Regarding the AdResS regions, we follow the methodology given in [3], which provides an equation to calculate the expected speedup in terms of the size of the FP and HY regions. This is viewed in terms of a reference, FP-only simulation, whose runtime t_{FP} is compared against that of the AdResS simulation t_{AdResS} as $t_{\mathrm{FP}}/t_{\mathrm{AdResS}}$. For this purpose, we make use of the generalized equation for the AdResS speedup, given in [3] as the inverse of

$$t_{\mathrm{AdResS}}/t_{\mathrm{FP}} = v + \left(1 + \frac{d_{\mathrm{HY}}}{d_{\mathrm{TOT}}} - v\right)\left(\frac{t_{\mathrm{F,FP}}/t_{\mathrm{FP}}}{P^2} + 1 - t_{\mathrm{F,FP}}/t_{\mathrm{FP}}\right), \qquad (6)$$

$$v = \frac{d_{\mathrm{FP}} + d_{\mathrm{HY}}}{d_{\mathrm{TOT}}} \quad , \quad S_{\mathrm{theo}} := \frac{1}{t_{\mathrm{AdResS}}/t_{\mathrm{FP}}}$$

the portion of the domain occupied by the HY and FP regions, where multi-site forces need to be computed; and $d_{\mathrm{HY,FP,TOT}}$ the length of the HY, FP, and domain, respectively, in the x-axis. The variable P stands for the total number of sites which will be coarsened in the CG region, e.g. four for a tetrahedral molecule. Moreover, eq. (6) does not depend on the absolute simulation time, but on the ratio between the time spent computing forces $t_{\mathrm{F,FP}}$ and the total time t_{FP}. In this way, we can measure the performance of each traversal given in fig. 2 with respect to the theoretical speedup computed by the inverse of eq. (6), in terms of the used region configuration.

4 Results

4.1 Validation results

The simulations in this section were generated using the tetrahedral molecule scenario, as described in [11], using `ls1 mardyn` . This is a pure Lennard-Jones molecule with four sites, fixed bond lengths, and no intramolecular interactions. For all molecules, the Lennard-Jones parameters are $\epsilon = 0.01$ and $\sigma = 1.0$, and a site mass of $m = 1.0$. We attach no intentional physical meaning to these values, as the focus of our work lies on the computational and algorithmic aspects; however, these parameters are used for the reduced units. All the results were obtained in a cube-shaped domain with $a = 36.845$ and a total of $N = 5324$ molecules at $T^* = 1.0$, which includes the artificially added particles due to halo cells. These halo, or ghost, cells are used for the implementation of the periodic boundary conditions. Therefore, the number density is approximately $\rho_N \approx 0.106$. As previously mentioned, these experiments were carried out with the linked cells data structure, for which the cell size and cutoff radius are set to $r_c = 3.0$. This allows for several cells to be located in each resolution region. The show correlation functions were profiled with the binning method, using 200 bins. Only Lennard-Jones reduced units are specified in all cases.

Our first results are related to the validation of the implemented IBI and AdResS functionalities, from sections 2 and 3 respectively. Based on the scheme presented in section 3, after obtaining an equilibrated phase space, the effective potential was computed via a completely coarsened domain. As previously

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

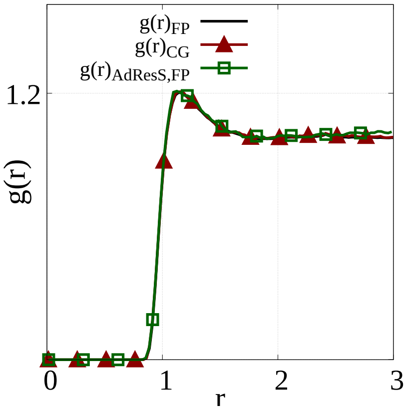


Fig. 3. Reference radial distribution function (solid line), $g_{\mathrm{ref}}(r)$, for a four-site LJ molecule at $T^* = 1$, compared against the RDF obtained with the computed effective potential (triangular points) using IBI in algorithm 1. The RDF generated by an AdResS-enabled simulation with $d_{\mathrm{FP}} = 10$ reproduces the reference structure (dashed line).

mentioned, the coarse molecules consist of a single site. The IBI method in algorithm 1 was configured with $N_{\mathrm{IBI}} = 10^3$ simulation steps between each potential update, see eq. (4). In all cases $\alpha = 1.0$, as in the single-state IBI in [6]. Additionally, the algorithm converged when the total simulation steps, $N = 10^5$, were reached or when a similarity of $\epsilon_{IBI} = 0.9970$ was reached. In our implementation, a decrease of convergence was also considered as a stopping condition. The value for ϵ_{IBI} was chosen based on the results given in [6].

A comparison between the reference RDF, and RDFs generated in coarsened phase space and an AdResS simulation, is shown in fig. 3. In `ls1 mardyn` , the cell traversal limits the found pairs to the cutoff radius, hence the computation of the RDF is limited to pairs within r_c. In the figure, it can be seen that a simulation using only coarsened molecules and the computed effective potential, can correctly reproduce the reference RDF to the predetermined tolerance ϵ_{IBI}. This allows us to use this potential for CG interactions in an AdResS-enabled simulation, which is shown in fig. 3 as the RDF for only the FP region with $d_{\mathrm{FP}} = 10$. The latter was averaged for $3 \cdot 10^4$ simulation steps.

Additional validation results are given in fig. 4. The IBI convergence is shown for the coarse-grained domain. It shows convergence after seven potential updates. Since the reference RDF is generated in a previous phase, the stride N_{IBI} allows for an averaged $g_i(r)$ to be obtained before the application of eq. (4), which reduces the error introduced by the instantaneous sampling of the correlation function. The effective potential after seven corrections of the initial guess shows signs of the sampling noise. This could be improved on by filtering,

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

as in [13], or by a larger N_{IBI}. However, we decided to use the data without filtering since no negative impact was observed from the noise.

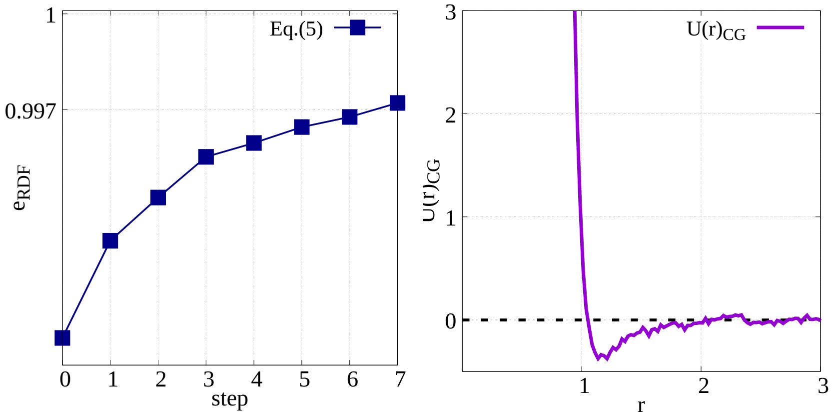


Fig. 4. (left)IBI convergence with $N_{\text{IBI}} = 10^3$ simulation steps between potential updates. (right) The effective potential, whose corresponding intermolecular forces reproduce $g_{\text{ref}}(r)$.

Artificial density gradients have been reported between the interfaces of the resolution regions. For our implementation, these are shown in fig. 5 for the number density across the whole domain. It is possible to correct them by the addition of a thermodynamic force, F_{th}, as in [9]. Since we follow the methodology given in [3], we decided not to include this correction in our experiments yet.

These density gradients consist of deviations from the bulk density, ρ_0, which are introduced by the lack of thermodynamic balance between the HY with the CG and FP regions, e.g. [11, 9, 17]. In fig. 5, the values have been normalized by ρ_0, and the bulk density is shown as a reference. Although the profile presents some differences with what is reported for this case study in the literature, [11, 10, 4], it is not the focus here to test the results of AdResS for complete physical correctness, but to study its parallel properties in `ls1 mardyn` . Thus, we assume that the validation results provided in this section attest to a sufficiently working implementation for the purposes of this work.

4.2 AdResS Speedup

In this section, we draw a comparison between the different `ls1 mardyn` linked cells traversals, described in section 3.2. Our analysis is based on the AdResS performance metrics described in section 2 and based on [3]. With these metrics we attempt to evaluate how AdResS impacts the performance of the simulation for each traversal in `ls1 mardyn` , as well as how these traversals affect the expected speedup with respect to an FP reference simulation. As benchmarking

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

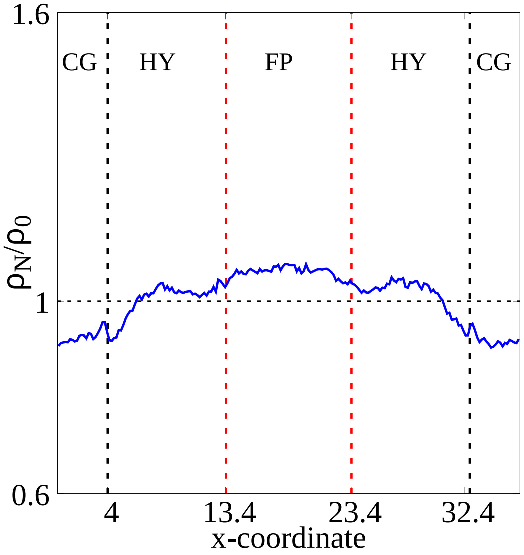


Fig. 5. Normalized number density profile, $\rho_N(r)$ for an AdResS-enabled simulation using a tetrahedral-molecule with Lennard-Jones sites. A single-site at the COM was used for the coarse-grain molecular model in the CG regions. A 100k time step average is shown.

scenarios, we use the tetrahedral molecule described in section 4.1 and compare it to a scenario based on the tetrahedral molecule previously used together with a domain with dimensions $(432 \times 48 \times 48)$. This allows us to compare the performance of AdResS in `ls1 mardyn` at increasing computational capacities, since the larger domain is run at node saturation.

Table 1. Total runtime, $\boldsymbol{t_{\mathrm{FP}}}$, of each tested traversal using an FP-only tetrahedral molecule scenario, with $\boldsymbol{t_{\mathrm{F,FP}}}$ the corresponding runtime spent on force computation. The theoretical speedup of (6) uses these values as a a base for the estimation. Results are shown for both the rectangular and cubic domains.

Domain size	$(432, 48, 48)$		$(36.845, 36.845, 36.845)$	
Traversal	$\boldsymbol{t_{\mathrm{FP}}}$	$\boldsymbol{t_{\mathrm{F,FP}}/t_{\mathrm{FP}}}$	$\boldsymbol{t_{\mathrm{FP}}}$	$\boldsymbol{t_{\mathrm{F,FP}}/t_{\mathrm{FP}}}$
`c04`	3.7987	0.6163	4.5375	0.8258
`sliced`	3.8246	0.6129	10.19	0.8366
`c08`	4.2436	0.6621	2.4067	0.7824

For every traversal, simulations with varying FP and HY region widths, that is d_{FP} and d_{HY}, were performed in the `HSuper` [4] distributed memory system.

---

[4] https://portal.hpc.hsu-hh.de/

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

Each `HSuper` node is equipped with a dual-socket Intel Icelake CPU, featuring a total of 2x36 compute cores and 256 GB RAM. We focus on node-level performance and therefore conducted single-node experiments. Every individual scenario was measured five times, for 250 simulation steps, and averaged, to account for computational jitter between production runs. As mentioned in section 3.2, every traversal is first benchmarked in an all-FP simulation from which t_{FP} is obtained. This time is then used for the theoretical speedup computation, see eq. (6), whereas the real speedup S_{sim} is measured as

$$S_{\mathrm{sim}} = t_{\mathrm{FP}} / t_{\mathrm{AdResS}}, \tag{7}$$

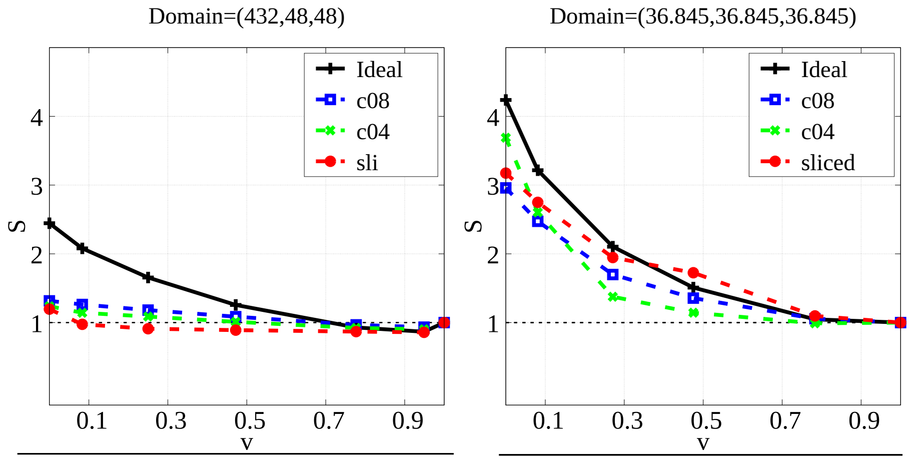
with t_{AdResS} the runtime measured from a simulation configured with $d_{\mathrm{FP}} \neq 0$ and $d_{\mathrm{HY}} \neq 0$. For our experiments, the ratio between d_{HY} and d_{FP} was chosen heuristically and only for performance measurement purposes. The underlying physical of aspects of these dimensions, were not accounted for in these simulation runs.

In table 1, the average wallclock times for every traversal is presented. The traversals for the rectangular prism domain were tested using a full `HSuper` node, i.e. 72 cores. On the other hand, the cubic domain was tested with 32 cores for the `c04` and `c08` traversals, and 6 cores for the `sliced`. In both scenarios, it was attempted to create slices of a single resolution, when using the `sliced` traversal. Additionally, the performance metrics are in terms of the FP times, not in terms of the absolute runtime. The ratio of time spent computing the short-range forces to the total simulation time, $t_{\mathrm{F,FP}}/t_{\mathrm{FP}}$, is given in the second column for each domain. This ratio is required in eq. (6), and illustrates further differences in performance within the `ls1 mardyn` implementation. It is seen that at node saturation, the `c04` traversal perform better than the others in terms of the overall time. Nevertheless, for a smaller computational load on the node there is a better performance in terms of $t_{\mathrm{F,FP}}/t_{\mathrm{FP}}$, independently of the runtimes.

We present speedup results each traversal in fig. 6, comparing the expected against the measured speedup for each traversal in both scenarios. As mentioned before, several widths for the FP and HY region were tested for vertical interfaces inside the simulation domain. Differences in the node-level performance are visible, between each domain size. In the rectangular domain, it is seen that for larger values of d_{FP}, the theoretical speedup is below the reference time. This can be related to the presence of large HY regions, which imply the computation of multi-site forces across a larger portion of the volume. Therefore more multi- and single-site forces must be computed. On the other hand, the cubic domain performs closer to the predicted speedup, however there is a loss of performance with respect to S_{theo}. Additionally, for both scenarios there is the AdResS-introduced overhead that could slow down the simulations. One of the slow-down factors is that the potential was generated using IBI in algorithm 1, which was based on a bin-based tabulation of the RDF, therefore the potential and forces are also tabulated instead of computed from a hard-coded analytical expression as is the case for the Lennard-Jones potential. Another AdResS-specific implementation

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

factor in `ls1 mardyn` , which could have decreased the speedup, is the fact that the resolution of every molecule is checked at the end of every step. Moreover, the force computation requires the previous determination of the interacting resolutions, thus adding a previous step for every pairwise interaction. Nevertheless, for small-sized FP and HY regions, there was a speedup with respect to the FP-only simulation. And for the large values of v, the real speedup follows the theoretical one.



d_{FP}	v	S_{theo} c08		c04		S_{sim} sliced	
0	0	2.6369	1.3160	2.3682	1.2378	2.3506	1.1956
12	0.0833	2.2123	1.2634	2.0249	1.1426	2.0214	0.9747
60	0.2546	1.7291	1.1818	1.6245	1.0877	1.6173	0.9106
90	0.4722	1.2948	1.0857	1.2401	1.0108	1.2363	0.8902
116	0.7778	0.9481	0.9668	0.9206	0.9223	0.9187	0.8688
216	0.9491	0.8782	0.9338	0.8620	0.9031	0.8608	0.8580
$\epsilon_{\mathrm{rms,theo}}$		0.6538		0.5854		0.6582	
$\epsilon_{\mathrm{rms,ideal}}$		0.5606		0.6233		0.7045	

d_{FP}	v	S_{theo} c08		c04		S_{sim} sliced	
0	0	3.753	2.959	4.429	3.689	4.646	3.174
1	0.081	2.936	2.473	3.321	2.603	3.433	2.749
5	0.271	1.993	1.699	2.143	1.376	2.184	1.946
7.5	0.475	1.455	1.356	1.527	1.144	1.546	1.725
10	0.783	1.024	1.056	1.056	0.989	1.064	1.097
$\epsilon_{\mathrm{rms,theo}}$		0.3863		0.5482		0.6703	
$\epsilon_{\mathrm{rms,ideal}}$		0.6294		0.4732		0.4872	

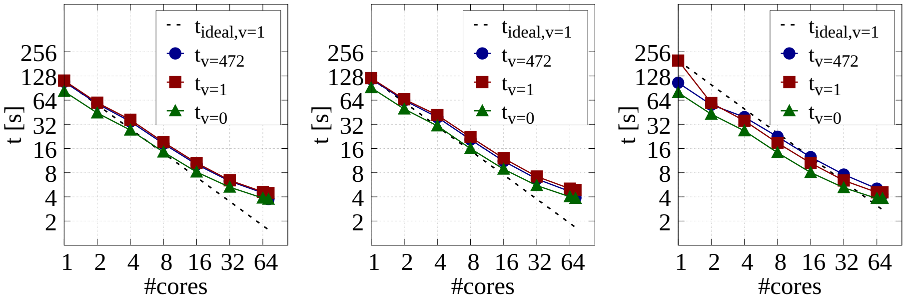
Fig. 6. (top) The computed speedup, S_{sim}, of the tested linked cell traversing routines. (bottom) Speedup data for increasing v, with $d_{\mathrm{HY}} = v\, d_{\mathrm{TOT}} - d_{\mathrm{FP}}$. Where $d_{\mathrm{TOT}} = 432$ and $d_{\mathrm{TOT}} = 36.845$, respectively for each scenario. For each traversal, the left column gives the theoretical speedup S_{theo}, see eq. (6), based on the respective reference t_{FP} in table 1. The right column shows the measured speedup S_{sim} computed using the runtimes measured in an AdResS simulation with the specified resolution regions in the leftmost columns, d_{FP} and v, the RMSE with respect to the ideal scenario and the expected speedups in fig. 6 is given in the last rows of each table. Fo

In addition to the speedup per traversal, an inter-traversal comparison is given with respect to an ideal scenario in fig. 6. This is motivated by the fact that the speedup values in fig. 6 and the runtimes in table 1 do not compare the traversals with each other, but against the performance in terms of t_{FP}. Therefore, we propose to use the average of $t_{\mathrm{F,FP}}/t_{\mathrm{FP}}$ for all our measurements

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

to create an ideal FP scenario with $t_{\mathrm{F,FP}} = 0.6304$ for the larger domain and $t_{\mathrm{F,FP}} = 0.815$ for the smaller, and compute the theoretical speedup for the tested values of v. This is plotted in fig. 6 for each domain, together with the RMS errors with respect to the ideal scenario and the speedup computed using eq. (6), $\epsilon_{\mathrm{rms,ideal}}$ and $\epsilon_{\mathrm{rms,eq.\ (6)}}$ respectively, are provided. It is seen that the performance of the traversals is negatively affected by the larger x-dimension in the rectangular domain.

Finally, a strong scalability study was carried out in `HSuper`for up to 72 cores for the $(432, 48, 48)$ domain size. The runtimes for 250 simulation steps are presented in fig. 7. For each of the traversals, the parallel speedup was measured for $v := (0, 0.472, 1)$. This allows for a comparison between the upper and lower limit scenarios in terms of size of the FP region and a mixed scenario with CG molecules in approximately half the domain. It is seen that for the lower core counts in the `sliced` traversal, the runtime of the AdResS simulation with $v = 0.472$ lies between the limit cases as expected. However, for `c04` and `c08` the runtimes of $v = 0$ are not significantly lower with respect to $t_{\mathrm{v=1}}$, which was also shown for the AdResS speedup study in fig. 6. Still, overall good node-level scalability is observed, and run times follow the ideal scaling curve rather well.



(a) `c04` (b) `c08` (c) `sliced`

Fig. 7. Strong scaling runtimes, in seconds, measured for all traversals over 250 simulation steps for the $(432, 48, 48)$ domain size. Times for $v := (0, 0.472, 1)$ are shown for up to 72 cores. The ideal time t_{ideal} is given with respect to the reference run where $v = 1.0$.

5 Conclusion

In this work, we describe and analyze the implementation of AdResS in `ls1 mardyn` . Introductions to AdResS in molecular dynamics and to IBI for the computation of the coarse-grain effective potential were given. Validation results, generated using the well-known tetrahedral molecule case study, show that our implementation can reproduce literature results with minor differences.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

The performance is analyzed in terms of the used geometry, i.e. the size of the resolution regions, d_{FP} and d_{HY} in terms v in eq. (6), where the multi-site interaction forces need to be computed. For this purpose, the speedup was measured with respect to a reference simulation with only FP molecules, using two different domain configurations. The results show that for a simulation run not at node saturation, the measured speedups are slightly below the theoretical values, as expected due to implementation overheads. And that at node saturation, a non-cubic domain in an AdResS-enabled simulation, presents performance losses for increasing v values, which requires further investigation. Nevertheless, it is shown that the difference between the measured and expected values is dependent on the used traversal, as well as on the resolution region configuration. It is seen that for some cases, the measured performance was better than the expected. An ideal scenario was constructed based on the reference performance ratio. It was used to compare the traversals to each other. For the cubic domain the measured speedup follows the theoretical closely, and the results of the rectangular domain show that none of the traversals could reach the theoretical values for the lower v. Finally, the strong scalability study for the $(432, 48, 48)$ domain shows that the traversals do not generally follow the predicted runtime decrease with respect to a full particle simulation and theoretical AdResS speedup as predicted by eq. (6).

These results will be taken as a starting point for further testing of AdResS in `ls1 mardyn` within the context of domain decomposition using distributed-memory systems.

Acknowledgments. Computational resources (HPC cluster `HSuper`) have been provided by the project hpc.bw, funded by dtec.bw – Digitalization and Technology Research Center of the Bundeswehr. dtec.bw is funded by the European Union – NextGenerationEU. The project has been supported by the BMBF project 3xa, 16ME0653, which has been financed by the European Union.

Disclosure of Interests. The authors have no competing interests to declare that are

References

1. Goga, N., Melo, M.N., Rzepiela, A.J., de Vries, A.H., Hadar, A., Marrink, S.J., Berendsen, H.J.C.: Benchmark of schemes for multiscale molecular dynamics simulations. Journal of Chemical Theory and Computation **11**(4), 1389–1398 (Apr 2015). https://doi.org/10.1021/ct501102b
2. Gratl, F.A., Seckler, S., Bungartz, H.J., Neumann, P.: N ways to simulate short-range particle systems: Automated algorithm selection with the node-level library autopas. Computer Physics Communications **273**, 108262 (2022). https://doi.org/10.1016/j.cpc.2021.108262
3. Junghans, C., Agarwal, A., Delle Site, L.: Computational efficiency and amdahl's law for the adaptive resolution simulation technique. Computer Physics Communications **215**, 20–25 (2017). https://doi.org/10.1016/j.cpc.2017.01.030

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4

4. Junghans, C., Poblete, S.: A reference implementation of the adaptive resolution scheme in espresso. Computer Physics Communications **181**(8), 1449–1454 (2010). https://doi.org/10.1016/j.cpc.2010.04.013
5. Krekeler, C., Agarwal, A., Junghans, C., Praprotnik, M., Delle Site, L.: Adaptive resolution molecular dynamics technique: Down to the essential. The Journal of Chemical Physics **149**(2), 024104 (07 2018). https://doi.org/10.1063/1.5031206
6. Moore, T.C., Iacovella, C.R., McCabe, C.: Derivation of coarse-grained potentials via multistate iterative boltzmann inversion. The Journal of Chemical Physics **140**(22), 224104 (06 2014). https://doi.org/10.1063/1.4880555
7. Neumann, P., Tchipev, N., Seckler, S., Heinen, M., Vrabec, J., Bungartz, H.J.: Petaflop molecular dynamics for engineering applications. In: Nagel, W.E., Kröner, D.H., Resch, M.M. (eds.) High Performance Computing in Science and Engineering ' 18. pp. 397–407. Springer International Publishing, Cham (2019)
8. Niethammer, C., Becker, S., Bernreuther, M., Buchholz, M., Eckhardt, W., Heinecke, A., Werth, S., Bungartz, H.J., Glass, C.W., Hasse, H., Vrabec, J., Horsch, M.: ls1 mardyn: The massively parallel molecular dynamics code for large systems. Journal of Chemical Theory and Computation **10**(10), 4455–4464 (2014). https://doi.org/10.1021/ct500169q, pMID: 26588142
9. Poblete, S., Praprotnik, M., Kremer, K., Delle Site, L.: Coupling different levels of resolution in molecular simulations. The Journal of Chemical Physics **132**(11), 114101 (03 2010). https://doi.org/10.1063/1.3357982
10. Potestio, R., Peter, C., Kremer, K.: Computer simulations of soft matter: Linking the scales. Entropy **16**(8), 4199–4245 (2014). https://doi.org/10.3390/e16084199
11. Praprotnik, M., Delle Site, L., Kremer, K.: Adaptive resolution molecular-dynamics simulation: Changing the degrees of freedom on the fly. The Journal of Chemical Physics **123**(22), 224106 (12 2005). https://doi.org/10.1063/1.2132286
12. Praprotnik, M., Matysiak, S., Site, L.D., Kremer, K., Clementi, C.: Adaptive resolution simulation of liquid water. Journal of Physics: Condensed Matter **19**(29), 292201 (jul 2007). https://doi.org/10.1088/0953-8984/19/29/292201
13. Reith, D., Pütz, M., Müller-Plathe, F.: Deriving effective mesoscale potentials from atomistic simulations. Journal of Computational Chemistry **24**(13), 1624–1636 (2003). https://doi.org/10.1002/jcc.10307
14. Rosenberger, D., Hanke, M., van der Vegt, N.F.: Comparison of iterative inverse coarse-graining methods. The European Physical Journal Special Topics **225**(8), 1323–1345 (Oct 2016). https://doi.org/10.1140/epjst/e2016-60120-1
15. Tchipev, N., Seckler, S., Heinen, M., Vrabec, J., Gratl, F., Horsch, M., Bernreuther, M., Glass, C.W., Niethammer, C., Hammer, N., Krischok, B., Resch, M., Kranzlmüller, D., Hasse, H., Bungartz, H.J., Neumann, P.: Twetris: Twenty trillion-atom simulation. The International Journal of High Performance Computing Applications **33**(5), 838–854 (2019). https://doi.org/10.1177/1094342018819741
16. Tchipev, N., Wafai, A., Glass, C.W., Eckhardt, W., Heinecke, A., Bungartz, H.J., Neumann, P.: Optimized force calculation in molecular dynamics simulations for the intel xeon phi. In: Hunold, S., Costan, A., Giménez, D., Iosup, A., Ricci, L., Gómez Requena, M.E., Scarano, V., Varbanescu, A.L., Scott, S.L., Lankes, S., Weidendorfer, J., Alexander, M. (eds.) Euro-Par 2015: Parallel Processing Workshops. pp. 774–785. Springer International Publishing, Cham (2015)
17. Wang, H., Schütte, C., Delle Site, L.: Adaptive resolution simulation (adress): A smooth thermodynamic and structural transition from atomistic to coarse grained resolution and vice versa in a grand canonical fashion. Journal of Chemical Theory and Computation **8**(8), 2878–2887 (2012). https://doi.org/10.1021/ct3003354, pMID: 26592127

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97570-7_4